

Systemet vi har konstruerat är en lösning inriktad på hantering av temperaturdata från simulerade IoT-enheter. Genom att använda den moderna SignalR-tekniken kan systemet realtidshantera data och kommunikation mellan dess komponenter.

Vår lösning är ett system som är designat för att hantera temperaturdata från simulerade IoT-enheter. När denna data kommer in till vår Web API server används SignalR för att distribuera datan till alla prenumererande webbklienter i realtid.

- **Server-del:** Serverdelen är ett .NET Core Web API och fungerar som en central knutpunkt för både insamling av data från IoT-enheter och distribution av den datan till webbklienter. HTTPS används för vanliga API anrop och SignalR används för att omedelbart notifiera enheter när ny temperaturinformation kommer in. När en IoT-enhet rapporterar en ny temperatur, lagras detta värde i vår databas och sedan skickas det vidare till alla prenumererande webbklienter.
- **Simulerad IoT-enhet (Konsolapplikation):** Denna komponent representerar en simulerad IoT-enhet. Genom HTTPS rapporterar den regelbundet in ett slumpmässigt genererat temperaturvärde till vår server var 5:e sekund. För att uppfylla säkerhetskraven krypterar den temperaturvärdet innan det skickas till serverdelen, vilket sedan dekrypteras av serverdelen innan det lagras.
- **Webbapplikation:** Webbapplikationen är en enkel .NET Core Web App som använder sig av static files. Denna prenumererar på uppdateringar av temperaturdata från serverdelen via SignalR. När en ny temperaturrapport tas emot av serverdelen, skickas den direkt till denna webbapplikation, som uppdaterar sitt visuella gränssnitt med den nya datan i realtid.
- **Säkerhet och autentisering:** För att uppnå högre säkerhetsnivåer, har systemet lagt till autentiseringsfunktioner på serverdelen. Detta möjliggörs med hjälp av JWT-tokens som tilldelas både IoT-enheter samt användare. Dessa tokens har specifika claims beroende på vilken typ det är, dvs IoT eller användare. Rapportering av ny temperaturdata kräver skriv-behörighet vilket nya IoT-enheter tilldelas då de registrerar sig vid uppstart av programmet.
- **Kommunikationsprotokoll:** SignalR används för realtidskommunikation mellan serverdelen och webbapplikationen för att tillåta uppdatering av gränssnittet i realtid. HTTPS används för vanliga API anrop både från den simulerade IoT-enheten när den ska posta ny temperaturdata, samt från vår webbapplikation då den exempelvis ska registrera nya användare, logga in/autentisera existerande användare och jwt tokens, hämta en inloggad användares roles samt logga ut en inloggad användare.

Genom att kombinera dessa komponenter har systemet skapat en robust och responsiv lösning för hantering av temperaturdata i realtid, med viktiga funktioner för säkerhet och dataskydd.

# C4 Container diagram Tobias Westman

